

IBM Tealeaf CX
Version 9 Release 0.2
June 18, 2015

Tealeaf Cookie Injector Manual



Note

Before using this information and the product it supports, read the information in “Notices” on page 29.

This edition applies to version 9, release 0, modification 2 of IBM Tealeaf CX and to all subsequent releases and modifications until otherwise indicated in new editions.

© **Copyright IBM Corporation 1999, 2015.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Tealeaf Cookie Injector Manual v

Chapter 1. Installing and Configuring the Tealeaf Cookie Injector 1

Overview	1
Supported Platforms.	2
SSL Support	2
Maintenance and Troubleshooting	2
Disable Other Sessioning Agents.	2
Acquiring Software	3
Installation and Configuration Instructions	3
Configuring Tealeaf Cookie Injector for Multiple Domains.	3
Configuring cxResults	3
Verifying the Installation	4
Upgrading the Tealeaf Cookie Injector	4
Upgrading Web Server	4
Backup	4
Managing Configuration Changes	5
Upgrade Process	5

Chapter 2. IBM Tealeaf Cookie Injector for Apache 1.3.x 7

System Requirements	7
Installing the Cookie Injector using a script	7
Installing the Cookie Injector manually	7
Configuration	8
Configuring for Multiple Domains	9

Chapter 3. IBM Tealeaf Cookie Injector for Apache 2.x 11

System Requirements	11
Installing the Cookie Injector using a script.	11
Installing the Cookie Injector manually	11
Configuration	12
Configuring for Multiple Domains.	13

Chapter 4. IBM Tealeaf Cookie Injector for Microsoft IIS 6.0 15

System Requirements	15
Installation Level	15
Server	15
Site	15
Using Setup Script	15
Manual Installation.	16
Configuration	16
Configuring for Multiple Domains.	17

Chapter 5. IBM Tealeaf Cookie Injector for Microsoft IIS 7.0 or Later 19

Installation	19
Installation Level	19
Using Setup Script	19
Manual Installation.	19
Configuration	20
Configuring for Multiple Domains.	21

Chapter 6. IBM Tealeaf Cookie Injector for SunOne-iPlanet 3.1 or later 23

System Requirements	23
Installing the Cookie Injector using a script.	23
Installing the Cookie Injector manually	23
Configuration	24
Configuring for Multiple Domains.	25

Chapter 7. IBM Tealeaf documentation and help 27

Notices 29

Trademarks	30
Privacy Policy Considerations	31

Tealeaf Cookie Injector Manual

The Tealeaf[®] Cookie Injector Manual provides installation and configuration instructions for the Tealeaf Cookie Injector, a lightweight cookie generator that can be installed in your web server environment. When the Cookie Injector is deployed, it generates session identifiers that are guaranteed to be unique within Tealeaf.

Note: The Cookie Injector is the preferred method of generating identifiers for Tealeaf. Tealeaf provides multiple mechanisms for getting session identifiers. See "Managing Data Sessionization in Tealeaf CX" in the *IBM Tealeaf CX Installation Manual*.

Chapter 1. Installing and Configuring the Tealeaf Cookie Injector

This document provides information on installation and configuration for the Tealeaf Cookie Injector on each supported platform.

Note: The Tealeaf Cookie Injector is an optional component separate from any other Tealeaf software you may have licensed. Your use of the Cookie Injector code is subject to the following terms:

THE COOKIE INJECTOR SOFTWARE IS PROVIDED BY TEALEAF TECHNOLOGY, INC., AN IBM® COMPANY ("TEALEAF") "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT ARE DISCLAIMED. IN NO EVENT SHALL TEALEAF OR ANY OF ITS PARENT, SUBSIDIARY OR AFFILIATE ENTITIES BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THE COOKIE INJECTOR SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Note: The Privacy and Electronic Communications Regulations 2003 contain specific directives regarding the use and management of cookies for European Union customers. You should consult your legal advisor regarding compliance with these directives and other applicable privacy related regulations.

Overview

The Tealeaf Cookie Injector is a lightweight platform-specific utility that is installed on the Web server or application server for the purpose of issuing HTTP cookies containing Globally Unique Identifiers (GUIDs). These GUIDs are used in sequencing (or 'sessionizing') the data captured by IBM Tealeaf cxImpact.

The Tealeaf Cookie Injector is used in conjunction with Tealeaf Passive Capture in cases where there is no existing cookie or application state variable that can be assigned as a reliable key for sessionizing the HTTP(S) request and response pairs into a logical visitor session.

The Tealeaf Cookie Injector is installed on each Web or application server, where it checks each incoming request for the configured cookie or cookies. If the request doesn't contain the cookie or cookies that the Cookie Injector is configured to issue, the Injector adds a Set-Cookie header to the response for each cookie.

- Optionally, it can also inject cookies into the request headers so it is seen by the components of the Web application that are downstream of the server where the Injector is installed.

The Cookie Injector can also add a response header that identifies the name of the server so that this information is available in hits captured by a IBM Tealeaf CX Passive Capture Application server.

- The Tealeaf Cookie Injector does not reconfigure or otherwise interfere with the ports in use by your web servers.

Supported Platforms

The Cookie Injector supports the following server platforms and operating systems:

Note: For unsupported operating systems and versions, source code can be provided upon request. For more information, please contact Tealeaf <http://support.tealeaf.com>.

Table 1. Supported Platforms

Server software	Windows 2000, 2003, 2008 and later	Linux (Intel), 6.2 kernel or later	Solaris (SPARC) 6 or later	IBM AIX® 4.3 or later
SunOne/iPlanet 3.1 or later	Yes	Yes	Yes	Yes
Apache 1.3.19 or later	Available on request	Yes	Yes	Yes
Apache 2.0.39 or later	Yes	Yes	Yes	Yes
MSFT IIS 5.0 or later	Yes	Not available	Not available	Not available

Note: Apache or SunOne/iPlanet filters may be available for other UNIX variants upon request. In cases where Tealeaf cannot provide a binary version, source code may be made available for customers to build their own.

Note: Cookie Injectors for earlier versions of Apache 1.3 available upon request.

SSL Support

The Tealeaf Cookie Injector leverages the native architecture of the Web server to evaluate and inject the configured Tealeaf cookie suite. The Cookie Injector executes this procedure after the Web server has decrypted or terminated the SSL connection. As a result, the Tealeaf Cookie Injector has no impact on the SSL termination process, and no considerations for SSL certificates are necessary.

Maintenance and Troubleshooting

The Tealeaf Cookie Injector should be maintained as a production configuration option of the Web server.

If you need troubleshoot a problem on the Web server, the Tealeaf Cookie Injector can be disabled, and Web server operations can be resumed. After the Cookie Injector is disabled, the Tealeaf cookies configured to be set by the Cookie Injector is no longer present in the HTTP(S) request and response data stream.

- The process to disable the Cookie Injector varies by platform.

Disable Other Sessioning Agents

For proper functioning of the Tealeaf Cookie Injector, other Tealeaf sessioning agents should be disabled, as they may override the values set by the Cookie Injector.

1. Through TMS, open the Pipeline Editor.

- See "Tealeaf Management System" in the *IBM Tealeaf cxImpact Administration Manual*.
- 2. The following session agents should not be in the main processing pipeline or any child pipeline off of the main processing pipeline:
 - [Sessioning]
 - [TLSessioning]
- 3. Through the PCA, you must disable sessioning. On the Pipeline tab in the PCA Web Console, verify that Use Sessioning is set to false. If you must change the setting, select **Save changes** at the bottom of the Pipeline tab page.
 - See "PCA Web Console - Pipeline Tab" in the *IBM Tealeaf Passive Capture Application Manual*

Acquiring Software

The Tealeaf Cookie Injector software is available for installation from the Tealeaf distribution in the following directory: TealeafCX\Sample Code\cxCookieFilters

Installation and Configuration Instructions

For your web platform, please use the link below for your web server platform for instructions on how to install and configure the Tealeaf Cookie Injector:

- Chapter 6, "IBM Tealeaf Cookie Injector for SunOne-iPlanet 3.1 or later," on page 23
- Chapter 2, "IBM Tealeaf Cookie Injector for Apache 1.3.x," on page 7
- Chapter 3, "IBM Tealeaf Cookie Injector for Apache 2.x," on page 11
- Chapter 4, "IBM Tealeaf Cookie Injector for Microsoft IIS 6.0," on page 15
- Chapter 5, "IBM Tealeaf Cookie Injector for Microsoft IIS 7.0 or Later," on page 19

Configuring Tealeaf Cookie Injector for Multiple Domains

You can configure the Tealeaf Cookie Injector to issue session identifiers for multiple domains served by the same Apache web server.

Note: The Tealeaf Cookie Injector must be installed globally. Limiting the module to specific virtual hosts is not a supported solution.

In the `teacookies.conf` configuration file, verify that the following properties are set to the listed values:

```
TLCookieIssueSID=On
' TlCookieDomain is not set
TLAddHostCookieDomain=True
```

Note: Please note that the `TLCookieDomain` parameter is commented out.

When `TLAddHostCookieDomain=True`, the two second-level domains are issued distinct identifier cookies, due to their different domain names.

Configuring cxResults

If IBM Tealeaf cxResults has been installed and configured before the Tealeaf Cookie Injector's installation and use, IBM Tealeaf cxResults needs to be updated to reflect the new user identifier provided by the Cookie Injector.

- See "Initial cxResults Configuration" in the *IBM Tealeaf cxResults Administration Manual*

Verifying the Installation

To verify the proper installation of the Tealeaf Cookie Injector, use the following steps to create a Tealeaf session with the Tealeaf cookie inserted into the request and to view the session through IBM Tealeaf CX RealiTea Viewer or through Browser-Based Replay in the Tealeaf Portal.

1. Using your web browser, visit the web application that is monitored by Tealeaf. Navigate to a few pages to create a traceable session.
2. Through the Tealeaf Portal or IBM Tealeaf CX RealiTea Viewer, search for Tealeaf sessions that have occurred since you installed the Cookie Injector.

Note:

- The Tealeaf cookie is installed by the Cookie Injector for each visitor and is saved by the session.
- Depending on the session timeout settings, you may have to search for active sessions and completed sessions.

For more information on search in RTV, see "RealiTea Viewer - Searching Sessions" in the *IBM Tealeaf RealiTea Viewer User Manual*.

For more information on search in the Tealeaf Portal, see "Searching Session Data" in the *IBM Tealeaf cxImpact User Manual*.

3. Replay a found session.

For more information on replay in RTV, see "RealiTea Viewer - Main Window" in the *IBM Tealeaf RealiTea Viewer User Manual*.

For more information on replay in the Tealeaf Portal, see "CX Browser Based Replay" in the *IBM Tealeaf cxImpact User Manual*.

4. Examine the Request View of the first page of the session.
 - a. In the [env] section, check HTTP_SET_COOKIE. The value for TLTSID should be defined.
 - b. In the [iamie] section, the value for TLTSID should be the same as the value in the [env] section.
5. Look at the last page of the session. Verify that the value for TLTSID is the same in both request view sections.

Upgrading the Tealeaf Cookie Injector

Upgrading Web Server

If you are upgrading your Web Server, you may need to uninstall the Cookie Injector and reinstall. For example, if you're upgrading from Apache 1.3 to Apache 2.0, a different version of the Cookie Injector is required. Please review the appropriate version of the Cookie Injector documentation to manage the upgrade.

Note: To preserve your configuration settings, you may want to migrate the settings from your old version to your new version. See "Managing Configuration Changes" on page 5.

Backup

Before you begin the upgrade, please backup the Tealeaf Cookie Injector directory on your web server.

Managing Configuration Changes

The configuration files for the Cookie Injector do not often change. However, before you upgrade, you should apply the configuration changes from the old files to a copy of the new configuration files.

- You may find it easier to diff the files and apply any new or changed settings from the new file into the old one.

Save the modified new configuration file outside the Cookie Injector directory.

Upgrade Process

1. Verify that backup is complete.
2. Copy the new .dll or .so files into the Tealeaf Cookie Injector directory on your web server.
3. Copy in the modified configuration files into the Tealeaf Cookie Injector directory.
4. Restart the Web Server.
5. Verify that the Cookie Injector is working properly. See “Verifying the Installation” on page 4.

Chapter 2. IBM Tealeaf Cookie Injector for Apache 1.3.x

Before you begin, make sure that you have the appropriate Tealeaf Cookie Injector software.

See “Acquiring Software” on page 3.

See Chapter 1, “Installing and Configuring the Tealeaf Cookie Injector,” on page 1.

System Requirements

There are no additional system requirements other than the Tealeaf Cookie Injector files. Of the Tealeaf Cookie Injector files listed below, `mod_teacookies1-3.so` and `teacookies.conf` are required files that must be deployed on the Web server in the appropriate directory:

- `mod_teacookies1-3.so`: Tealeaf Cookie Injector file for Apache 1.3.x
- `teacookies.conf`: Configuration file for the Tealeaf Cookie Injector for Apache 1.3.x

Installing the Cookie Injector using a script

This section describes how to install the cookie injector using a script.

Use the included setup script to install the Cookie Injector. This script automatically updates the appropriate configuration files for Apache 1.3.x and copies the Cookie Injector files to the correct location.

1. Run the setup script, passing the full path to HTTPd as an argument. For example, `./setup /usr/local/apache/bin/httpd`.
2. Configure the Cookie Injector by editing `teacookies.conf`. See “Configuration” on page 8.
3. Run `httpd -t` to test the configuration, and verify that no errors are displayed.
4. Restart Apache to load the Cookie Injector.

Installing the Cookie Injector manually

This section describes how to manually install the Cookie Injector.

1. Copy `mod_teacookies1-3.so` and `teacookies.conf` from the distribution disk to a directory on a local storage drive on the web server.
2. Add the following line to the end of the Apache `httpd.conf` configuration file, replacing `<path-to-teacookies>` with the actual path where `teacookies.conf` is located. For example: `Include <path-to-teacookies>`
3. Configure the Cookie Injector by editing `teacookies.conf`. See “Configuration” on page 8.
4. Run `httpd -t` to test the configuration, and verify that no errors are displayed.
5. Restart Apache to load the Cookie Injector.

Configuration

This section contains configuration information for the Cookie Injector and Apache version 1.3.x

Table 2. Configuration settings for *teacookies.conf*.

Option	Description
LoadModule teacookies_module modules/mod_teacookies1-3.so	This line loads mod_teacookies1-3.so. In the command, replace modules with the actual path to mod_teacookies1-3.so.
TLCookieEnabled On	This option determines if the cookie filter is active or not. Setting this option to Off disables the cookie filter the next time Apache is restarted.
TLCookieIssueSID On	This option specifies when to add a Set-Cookie header to the response for TLTSID (session ID cookie) if this cookie is not present in the request. The default value for this option is On.
TLCookieIssueHID Off	This option specifies when to add a Set-Cookie header to the response for TLTHID (hit ID cookie). This header is added to every response if TLTCookieIssueHID=On. The default value for this option is Off.
TLCookieIssueUID On	This option specifies when to add a Set-Cookie header to the response for TLTUID (user ID cookie) if this cookie is not present in the request. The default value for this option is Off.
TLCookieInjectSID Off	This option specifies when to insert a new TLTSID (session ID cookie) into the Cookie request header if this cookie is not present in the request. The cookie header is added when none exists. The default value for this option is Off.
TLCookieInjectHID Off	This option specifies when to insert a new TLTHID (hit ID cookie) into the Cookie request header (Cookie header is added if none exists). The default value for this option is Off.
TLCookieInjectUID Off	This option specifies when to insert a new TLTUID (user ID cookie) into the Cookie request header if this cookie is not present in the request (Cookie header is added if none exists). The default value for this option is Off.
TLCookieDomain .mydomain.com	This option sets the domain added to issued cookies. The setting begins with a period and contain at least two parts, which would work for any subdomain of the specified domain. For example, www.mydomain.com and www2.mydomain.com. If it is not specified, then the domain is parsed from the Host header in the request.
TLCookieAddHostCookieDomain On	If set to True, then the domain that is used for issued cookies is parsed from the Host header in the request. This option is useful for sites that have more than one top-level domain (for example, www.mystuff.com and www.foobar.com). This option is ignored if there is a value that is specified for TLTCookieDomain. The default value for this option is On.

Table 2. Configuration settings for *teacookies.conf*. (continued)

Option	Description
<code>TLCookieExtendedCookieDomain Off</code>	If set to <code>True</code> (when <code>AddHostCookieDomain=True</code>), then the domain for Set-Cookies headers is set to the last two parts of the host name (from the Host header) if the last part is <code>.com</code> , <code>.mil</code> , <code>.gov</code> , <code>.edu</code> , <code>.org</code> , <code>.net</code> , or <code>.int</code> , or the last three parts of the Host if not. When set to <code>False</code> (the default), the domain for Set-Cookies is set to the entire domain name from the Host header, starting from the first period.
<code>TLCookieSecureCookie Off</code>	This option specifies when to add the Secure attribute to Set-Cookie headers when setting cookies. Setting this option to <code>True</code> can cause the browser to return the Tealeaf cookies for HTTPS requests only. The default value for this option is <code>Off</code> .
<code>TLCookieIssueHostName Off</code>	This option specifies when to add a host name header to the response that identifies the web server. The default value for this option is <code>Off</code> .
<code>TLCookieHostName=<machinename></code>	<code>TLCookieHostName</code> specifies the name of the web server to be returned in the host name response header (if <code>TLCookieIssueHostName=On</code>). If the value of <code>HostName</code> is set to default, then the computer name is used. The default value for this option is <code>default</code> .

Configuring for Multiple Domains

If you are configuring the Tealeaf Cookie Injector to issue identifiers for multiple domains, additional configuration is required.

- See Chapter 1, “Installing and Configuring the Tealeaf Cookie Injector,” on page 1.

Chapter 3. IBM Tealeaf Cookie Injector for Apache 2.x

Before you begin, make sure that you have the appropriate Tealeaf Cookie Injector software.

See “Acquiring Software” on page 3.

See Chapter 1, “Installing and Configuring the Tealeaf Cookie Injector,” on page 1.

System Requirements

There are no additional system requirements other than the Tealeaf Cookie Injector files. Of the Tealeaf Cookie Injector files listed below, `mod_teacookies.so` and `teacookies.conf` are required files that must be deployed on the Web server in the appropriate directory:

- `mod_teacookies.so`: Tealeaf Cookie Injector file for Apache 2.x
- `teacookies.conf`: Configuration file for the Tealeaf Cookie Injector for Apache 2.x

Installing the Cookie Injector using a script

This section describes how to install the cookie injector using a script.

Use the included setup script to install the Cookie Injector. This script automatically updates the appropriate configuration files for Apache 2.x and copies the Cookie Injector files to the correct location.

1. Run the setup script, passing the full path to HTTPd as an argument. For example, `./setup /usr/local/apache/bin/httpd`.
2. Configure the Cookie Injector by editing `teacookies.conf`. See “Configuration” on page 12.
3. Run `httpd -t` to test the configuration, and verify that no errors are displayed.
4. Restart Apache to load the Cookie Injector.

Installing the Cookie Injector manually

This section describes how to manually install the Cookie Injector.

1. Copy `mod_teacookies.so` and `teacookies.conf` from the distribution disk to a directory on a local storage drive on the web server.
2. Add the following line to the end of the Apache configuration file, replacing `<path-to-teacookies>` with the actual path where `teacookies.conf` is located. For example: `Include path-to-teacookies/teacookies.conf`
3. Configure the cookie filter by editing `teacookies.conf`. See “Configuration” on page 12.
4. Run `httpd -t` to test the configuration, and verify that no errors are displayed.
5. Restart Apache to load the Cookie Injector.

Configuration

This section contains configuration information for the Cookie Injector and Apache version 2.x

Table 3. Configuration settings for *teacookies.conf*.

Option	Description
LoadModule teacookies_module modules/mod_teacookies.so	This line loads <code>mod_teacookies.so</code> . In the command, replace <code>modules</code> with the actual path to <code>mod_teacookies.so</code> .
TLCookieEnabled On	This option determines if the cookie filter is active or not. Setting this option to <code>Off</code> disables the cookie filter the next time Apache is restarted.
TLCookieIssueSID On	This option specifies when to add a Set-Cookie header to the response for <code>TLTSID</code> (session ID cookie) if this cookie is not present in the request. The default value for this option is <code>On</code> .
TLCookieIssueHID Off	This option specifies when to add a Set-Cookie header to the response for <code>TLTHID</code> (hit ID cookie). This header is added to every response if <code>TLCookieIssueHID=On</code> . The default value for this option is <code>Off</code> .
TLCookieIssueUID On	This option specifies when to add a Set-Cookie header to the response for <code>TLTUID</code> (user ID cookie) if this cookie is not present in the request. The default value for this option is <code>Off</code> .
TLCookieInjectSID Off	This option specifies when to insert a new <code>TLTSID</code> (session ID cookie) into the Cookie request header if this cookie is not present in the request. The cookie header is added when none exists. The default value for this option is <code>Off</code> .
TLCookieInjectHID Off	This option specifies when to insert a new <code>TLTHID</code> (hit ID cookie) into the Cookie request header (Cookie header is added if none exists). The default value for this option is <code>Off</code> .
TLCookieInjectUID Off	This option specifies when to insert a new <code>TLTUID</code> (user ID cookie) into the Cookie request header if this cookie is not present in the request (Cookie header is added if none exists). The default value for this option is <code>Off</code> .
TLCookieDomain .mydomain.com	This option sets the domain added to issued cookies. The setting begins with a period and contain at least two parts, which would work for any subdomain of the specified domain. For example, <code>www.mydomain.com</code> and <code>www2.mydomain.com</code> . If it is not specified, then the domain is parsed from the Host header in the request.
TLCookieAddHostCookieDomain On	If set to <code>True</code> , then the domain that is used for issued cookies is parsed from the Host header in the request. This option is useful for sites that have more than one top-level domain (for example, <code>www.mystuff.com</code> and <code>www.foobar.com</code>). This option is ignored if there is a value that is specified for <code>TLCookieDomain</code> . The default value for this option is <code>On</code> .

Table 3. Configuration settings for *teacookies.conf*. (continued)

Option	Description
<code>TLCookieExtendedCookieDomain Off</code>	If set to <code>True</code> (when <code>AddHostCookieDomain=True</code>), then the domain for Set-Cookies headers is set to the last two parts of the host name (from the Host header) if the last part is <code>.com</code> , <code>.mil</code> , <code>.gov</code> , <code>.edu</code> , <code>.org</code> , <code>.net</code> , or <code>.int</code> , or the last three parts of the Host if not. When set to <code>False</code> (the default), the domain for Set-Cookies is set to the entire domain name from the Host header, starting from the first period.
<code>TLCookieSecureCookie Off</code>	This option specifies when to add the Secure attribute to Set-Cookie headers when setting cookies. Setting this option to <code>True</code> can cause the browser to return the Tealeaf cookies for HTTPS requests only. The default value for this option is <code>Off</code> .
<code>TLCookieIssueHostName Off</code>	This option specifies when to add a host name header to the response that identifies the web server. The default value for this option is <code>Off</code> .
<code>TLCookieHostName=<machinename></code>	<code>TLCookieHostName</code> specifies the name of the web server to be returned in the host name response header (if <code>TLCookieIssueHostName=On</code>). If the value of <code>HostName</code> is set to default, then the computer name is used. The default value for this option is <code>default</code> .

Configuring for Multiple Domains

If you are configuring the Tealeaf Cookie Injector to issue identifiers for multiple domains, additional configuration is required.

- See Chapter 1, “Installing and Configuring the Tealeaf Cookie Injector,” on page 1.

Chapter 4. IBM Tealeaf Cookie Injector for Microsoft IIS 6.0

Before you begin, you must acquire the appropriate Tealeaf Cookie Injector software.

See Chapter 1, “Installing and Configuring the Tealeaf Cookie Injector,” on page 1.

This section describes how to install and configure the Tealeaf Cookie Injector for Microsoft IIS 6.0.

- For more information on the Tealeaf Cookie Injector for later versions, see Chapter 5, “IBM Tealeaf Cookie Injector for Microsoft IIS 7.0 or Later,” on page 19.

System Requirements

There are no additional system requirements other than the Tealeaf Cookie Injector files. The Tealeaf Cookie Injector files include the files listed below of which TeaCookiesIIS.dll and TeaCookiesIIS.cfg are required files that must be deployed on the Web server in the appropriate directory:

- TeaCookiesIIS.dll: Tealeaf Cookie Injector file for Microsoft IIS 6.0
- TeaCookiesIIS.cfg: Configuration file for the Tealeaf Cookie Injector for Microsoft IIS 6.0

Installation Level

Server

If you wish to apply the Tealeaf Cookie Injector to all sites hosted on the web server, you can install the software at the server level.

Make all configuration changes before you perform an IISReset.

Automated installation automatically installs at the server level. See “Using Setup Script.”

Site

If you only wish to apply the Tealeaf Cookie injector to some sites on the web server, the software must be installed in each site that you wish Tealeaf to capture.

Note: If you're installing at the site level on IIS 6 or higher, to activate Tealeaf Cookie Injector after it is installed, an IIS reset is not required. Instead, you may restart the affected sites or recycle the worker processes.

See “Manual Installation” on page 16.

Using Setup Script

It is recommended that you run setup.vbs to install the cookie injector. This script installs the cookie injector at the global level and updates the Windows registry with values needed to properly display event log messages. The steps are as follows:

1. Run setup.vbs and follow the instructions displayed.
2. Configure the cookie filter by editing TeaCookiesIIS.cfg as described below. See "Configuration."
3. Restart IIS to load the Cookie Injector.
4. **IIS 6.0 and higher:** After the IIS reset, use your web browser to visit the web site that Tealeaf is monitoring. When your initial request is made to the web server, a message is logged to the application event log, and the Priority for the Tealeaf Cookie Injector is updated in the IIS console.

Manual Installation

1. Copy TeaCookiesIIS.dll and TeaCookiesIIS.cfg from the distribution disk to a directory on a local hard disk on the Web server.
2. In Internet Services Manager, add TeaCookiesIIS.dll as an ISAPI filter at the desired level (i.e., global, Web site, application). If you are monitoring all sites on the Web server, it is recommended that you add the filter at the global level.
3. Configure the cookie filter by editing TeaCookiesIIS.cfg as described below. See "Configuration."
4. Restart IIS to load the Cookie Injector.

Configuration

The configuration of the IIS Cookie Injector is controlled by TeaCookiesIIS.cfg. The following options are available:

Option Description

Enabled=True

This option determines whether the Cookie Injector is active or not. Setting this option to False disables the Cookie Injector the next time IIS is restarted.

IssueSID=True

This option specifies whether to add a Set-Cookie header to the response for TLT SID (session ID cookie) if this cookie is not present in the request. The default value for this option is True.

IssueHID=False

This option specifies whether to add a Set-Cookie header to the response for TLTHID (hit ID cookie). This header is added to every response if IssueHID=True. The default value for this option is False.

IssueUID=True

This option specifies whether to add a Set-Cookie header to the response for TLTUID (user ID cookie) if this cookie is not present in the request. The default value for this option is False.

InjectSID=False

This option specifies whether to insert a new TLT SID (session ID cookie) into the Cookie request header if this cookie is not present in the request (Cookie header is added if none exists). The default value for this option is False.

InjectHID=False

This option specifies whether to insert a new TLTHID (hit ID cookie) into the Cookie request header (Cookie header is added if none exists). The default value for this option is False.

InjectUID=False

This option specifies whether to insert a new TLTUID (user ID cookie) into the Cookie request header if this cookie is not present in the request (Cookie header is added if none exists). The default value for this option is False.

FilterPriority=low

This option determines the priority level at which the filter is executed. This setting can be used to adjust the filter priority to avoid issues with other ISAPI filters. Possible values are low, medium or high. The default value for this option is low.

CookieDomain=.mydomain.com

This option sets the domain added to issued cookies. It should start with a dot and contain at least two parts, as shown in the above example, which would work for any sub-domain of the specified domain, such as www.mydomain.com and www2.mydomain.com. If it is not specified, then the domain is parsed from the Host header in the request.

AddHostCookieDomain=True

If set to True, then the domain used for issued cookies is parsed from the Host header in the request. This option is useful for sites that have more than one top-level domain (e.g., www.mystuff.com and www.foobar.com). This option is ignored if there is a value specified for CookieDomain. The default value for this option is True.

ExtendedCookieDomain=False

If set to True (when AddHostCookieDomain=True), then the domain for Set-Cookies headers is set to the last two parts of the Host name (from the Host header) if the last part is .com, .mil, .gov, .edu, .org, .net, or .int, or the last three parts of the Host if not. If set to False (the default), then the domain for Set-Cookies is set to the entire domain name from the Host header, starting from the first dot.

SecureCookie=False

This option specifies whether to add the Secure attribute to Set-Cookie headers when setting cookies. Note that setting this option to True causes in most cases the browser to only return the Tealeaf cookies for HTTPS requests. The default value for this option is False.

IssueHostName=False

This option specifies whether to add a HostName header to the response which identifies the Web server. The default value for this option is False.

HostName=<machinename>

HostName is used to specify the name of the Web server to be returned in the HostName response header (if IssueHostName=True). If the value of HostName is set to default, then the computer name is used. The default value for this option is default.

Configuring for Multiple Domains

If you are configuring the Tealeaf Cookie Injector to issue identifiers for multiple domains, additional configuration is required.

- See Chapter 1, “Installing and Configuring the Tealeaf Cookie Injector,” on page 1.

Chapter 5. IBM Tealeaf Cookie Injector for Microsoft IIS 7.0 or Later

This section describes how to install and configure the Tealeaf Cookie Injector for Microsoft IIS 7.0 or later.

Note: The Tealeaf Cookie Injector works with IIS 7.0 on Windows Server 2008 or later.

Before you begin, you must acquire the appropriate Tealeaf Cookie Injector software. See Chapter 1, “Installing and Configuring the Tealeaf Cookie Injector,” on page 1.

Installation

Installation Level

Server

If you wish to apply the Tealeaf Cookie Injector to all sites hosted on the web server, you can install the software at the server level.

Make all configuration changes before you perform an IISReset.

Automated installation automatically installs at the server level. See “Using Setup Script.”

Site

If you only wish to apply the Tealeaf Cookie injector to some sites on the web server, the software must be installed in each site that you wish Tealeaf to capture.

Note: If you are installing at the site level on IIS 6 or higher, to activate Tealeaf Cookie Injector after it is installed, an IIS reset is not required. Instead, you may restart the affected sites or recycle the worker processes.

See “Manual Installation.”

Using Setup Script

The included setup script (setup.cmd) can be used to install the TeaLeaf Cookie Injector module on IIS. This script installs the Cookie Injector at the global level.

The usage for the setup script is as follows:

```
setup.cmd [-u]
```

where:

-u = Uninstalls the Cookie Injector module.

Manual Installation

If you want to install the Cookie Injector only for specific web sites or just prefer to install manually, then use the following steps:

1. Copy TeaCookiesIIS7.dll and TeaCookiesIIS7.cfg to a directory on a local hard drive on the Web server.
2. In Internet Services Manager, add TeaCookiesIIS7.dll as a native module at the desired level (i.e., global, Web site, application).
 - If you are monitoring all sites on the Web server, it is recommended that you add the module at the global level.
3. To configure the Cookie Injector, edit TeaCookiesIIS7.cfg. See “Configuration.” Perform all configuration changes before initiating an IIS reset.
4. To load the newly installed and configured Cookie Injector, restart IIS.
5. After the IIS reset, use your web browser to visit the web site that Tealeaf is monitoring. When your initial request is made to the web server, a message is logged to the application event log, and the Priority for the Tealeaf Cookie Injector is updated in the IIS console.

Configuration

The configuration of the IIS 7 Cookie Injector is controlled by TeaCookiesIIS7.cfg. The following options are available:

Option Description

Enabled

This option determines whether the Cookie Injector is active or not. To disable the Cookie Injector, set this option to False and restart IIS.

- The default value is True.

IssueSID

This option specifies whether to add a Set-Cookie header to the response for TLTSID (session ID cookie) if this cookie is not present in the request.

- The default value is True.

IssueHID

This option specifies whether to add a Set-Cookie header to the response for TLTHID (hit ID cookie). This value is added to every response if IssueHID is set to True.

- The default value is False.

IssueUID

This option specifies whether to add a Set-Cookie header to the response for TLTUID (user ID cookie) if this cookie is not present in the request.

- The default value is False.

InjectSID

This option specifies whether to insert a new TLTSID (session ID cookie) into the Cookie request header if this cookie is not present in the request (Cookie header is added if none exists).

- The default value is False.

InjectHID

This option specifies whether to insert a new TLTHID (hit ID cookie) into the Cookie request header (Cookie header is added if none exists).

- The default value is False.

InjectUID

This option specifies whether to insert a new TLTUID (user ID cookie) into the Cookie request header if this cookie is not present in the request (Cookie header is added if none exists).

- The default value is False.

FilterPriority

This option determines the priority level at which the module executes. This setting can be used to adjust the module priority to avoid issues with other modules. Possible values are low, medium or high.

- The default value is low.

CookieDomain

This option sets the domain added to issued cookies. It should start with a dot and contain at least two parts (e.g., .mydomain.com). This example works for any sub-domain of the specified domain, such as www.mydomain.com and www2.mydomain.com.

- If this value is not specified, then the domain is parsed from the Host header in the request.

AddHostCookieDomain

If set to True, then the domain used for issued cookies is parsed from the Host header in the request. This setting is useful for sites that have more than one top-level domain (e.g., www.mystuff.com and www.foobar.com). This option is ignored if there is a value specified for CookieDomain.

- The default value is True.

ExtendedCookieDomain

If set to True and AddHostCookieDomain=True, then the domain for Set-Cookies headers is set to the last two parts of the Host name (from the Host header) if the last part is .com, .mil, .gov, .edu, .org, .net, or .int, or to the last three parts of the Host if not.

When set to False, the domain for Set-Cookies is set to the entire domain name from the Host header, starting from the first dot.

- The default value is False.

SecureCookie

This option specifies whether to add the Secure attribute to Set-Cookie headers when setting cookies.

Note: Setting this value to True in most cases causes the browser to only return the TeaLeaf cookies for HTTPS requests.

- The default value is False.

IssueHostName

This option specifies whether to add a HostName header to the response to identify the Web server.

- The default value is False.

HostName

This setting is used to specify the name of the Web server that is returned in the HostName response header (if IssueHostName=True). If the value of HostName is set to default, then the computer name is used.

- The default value is default.

Configuring for Multiple Domains

If you are configuring the Tealeaf Cookie Injector to issue identifiers for multiple domains, additional configuration is required.

- See “Configuring Tealeaf Cookie Injector for Multiple Domains” on page 3.

Chapter 6. IBM Tealeaf Cookie Injector for SunOne-iPlanet 3.1 or later

Before you begin, you must acquire the appropriate Tealeaf Cookie Injector software.

See Chapter 1, “Installing and Configuring the Tealeaf Cookie Injector,” on page 1.

System Requirements

There are no additional system requirements other than the Tealeaf Cookie Injector files.

Note: Of the Tealeaf Cookie Injector files listed below, `teacookies-iplanet.so` and `teacookies.conf` are required files that must be deployed on the web server in the appropriate directory.

The additional files are provided as configuration examples to aid the configuration process:

- `teacookies-iplanet.so`: Tealeaf Cookie Injector for Sun One/iPlanet.
- `teacookies.conf`: Configuration file for the Tealeaf Cookie Injector for Sun One/iPlanet.
- `obj.conf.add`: Sample configuration directives to add to the iPlanet `obj.conf` or `object.conf` file.
- `magnus.conf.add`: Sample configuration directives to add to `magnus.conf`, for later versions that include the `Init` directives in `magnus.conf`.

Installing the Cookie Injector using a script

This section describes how to install the cookie injector using a script.

Use the included setup script (`setup.vbs` for Windows) to install the Cookie Injector. This script automatically updates the appropriate configuration files for SunOne/iPlanet and copies the Cookie Injector files to the plugins directory.

1. Run the setup script `setup.vbs` and use the SunOne/iPlanet install directory as an argument. For example, `./setup /usr/local/iplanet`.
2. Configure the Cookie Injector by editing `teacookies.conf`. See “Configuration” on page 24.
3. Restart SunOne/iPlanet to load the Cookie Injector.

Installing the Cookie Injector manually

This section describes how to manually install the Cookie Injector.

1. Copy `teacookies-iplanet.so` and `teacookies.conf` from the distribution disk to a directory on a local storage drive on the web server.
2. If your version of SunOne/iPlanet has an `obj.conf` file in the config directory and the `Init` directives are included in `obj.conf`, then add the contents of `obj.conf.add` according to instructions in `obj.conf.add`. Make sure to change the paths of `teacookies-iplanet.so` and `teacookies.conf` in `obj.conf` to match the actual locations of the files.

3. If your version of SunOne/iPlanet does not have an `obj.conf` file in the `config` directory, or the `Init` directives are in `magnus.conf`; then, add the contents of `magnus.conf.add` and `obj.conf.add` to `magnus.conf` and `obj.conf` or `object.conf`, according to the comments in the `*.conf.add` files. Make sure to change the paths of `teacookies-iplanet.so` and `teacookies.conf` in the `.conf` files to match the actual locations of the files.
4. Configure the cookie filter by editing `teacookies.conf`. See “Configuration.”
5. Restart SunOne/iPlanet to load the Cookie Injector.

Configuration

This section contains configuration information for the Cookie Injector and SunOne/iPlanet version 3.1 or later.

Table 4. Configuration settings for `teacookies.conf`.

Option	Description
<code>Enabled=True</code>	This option determines if the Cookie Injector is active or not. Setting this option to <code>False</code> disables the Cookie Injector the next time the SunOne/iPlanet web server is restarted.
<code>NoCache=False</code>	Setting this option to <code>True</code> forces the elimination of <code>IsModified</code> -type checks that would return an HTTP 304 (Not Modified) response. These responses don't add the <code>Set-Cookie</code> headers issued by the Cookie Injector. This option should only be used if needed. The default value for this option is <code>False</code> .
<code>IssueSID=True</code>	This option specifies whether to add a <code>Set-Cookie</code> header to the response for <code>TLTSID</code> (session ID cookie) if this cookie is not present in the request. This cookie is intended to be issued once per web user session. The default value for this option is <code>True</code> .
<code>IssueHID=False</code>	This option specifies whether to add a <code>Set-Cookie</code> header to the response for <code>TLTHID</code> (hit ID cookie). This header is added to every response if <code>IssueHID=True</code> . The default value for this option is <code>False</code> .
<code>IssueUID=True</code>	This option specifies whether to add a <code>Set-Cookie</code> header to the response for <code>TLTUID</code> (user ID cookie) if this cookie is not present in the request. This cookie provides a permanent unique identifier for the web user. The default value for this option is <code>False</code> .
<code>InjectSID=False</code>	This option specifies whether to insert a new <code>TLTSID</code> (session ID cookie) into the Cookie request header if this cookie is not present in the request (the Cookie header is added if none exists). The default value for this option is <code>False</code> .
<code>InjectHID=False</code>	This option specifies whether to insert a new <code>TLTHID</code> (hit ID cookie) into the Cookie request header (the Cookie header is added if none exists). The default value for this option is <code>False</code> .

Table 4. Configuration settings for *teacookies.conf*. (continued)

Option	Description
CookieDomain=.mydomain.com	This option sets the domain added to issued cookies. It should start with a period and contain at least two parts which works for any sub domain of the specified domain, such as <code>www.mydomain.com</code> and <code>www2.mydomain.com</code> . If this is not specified, then the domain is parsed from the Host header in the request.
AddHostCookieDomain=True	If set to True, then the domain used for issued cookies is parsed from the Host header in the request. This option is useful for sites that have more than one top-level domain (e.g., <code>www.mystuff.com</code> and <code>www.foobar.com</code> would have their cookie domains set to <code>.mystuff.com</code> and <code>.foobar.com</code>). This option is ignored if there is a value specified for CookieDomain. The default value for this option is True.
ExtendedCookieDomain=False	If set to True (when AddHostCookieDomain=True), then the domain for Set-Cookies headers is set to the last two parts of the Host name (from the Host header) if the last part is <code>.com</code> , <code>.mil</code> , <code>.gov</code> , <code>.edu</code> , <code>.org</code> , <code>.net</code> , or <code>.int</code> , or the last three parts of the Host if not. If set to False (the default), then the domain for Set-Cookies is set to the entire domain name from the Host header, starting from the first period.
SecureCookie=False	This option specifies whether to add the Secure attribute to Set-Cookie headers when setting cookies. Setting this option to True can cause the browser to only return the Tealeaf cookies for HTTPS requests. The default value for this option is False.
IssueHostName=False	This option specifies whether to add a HostName header to the response to identify the Web server. The default value for this option is False.
HostName=<machinename>	HostName is used to specify the name to be returned in the HostName response header (if IssueHostName=True). If the value of HostName is set to default, then the DNS hostname of the computer is used. The default value for this option is default.

Configuring for Multiple Domains

If you are configuring the Tealeaf Cookie Injector to issue identifiers for multiple domains, additional configuration is required.

- See “Configuring Tealeaf Cookie Injector for Multiple Domains” on page 3.

Chapter 7. IBM Tealeaf documentation and help

IBM Tealeaf provides documentation and help for users, developers, and administrators.

Viewing product documentation

All IBM Tealeaf product documentation is available at the following website:

<https://tealeaf.support.ibmcloud.com/>

Use the information in the following table to view the product documentation for IBM Tealeaf:

Table 5. Getting help

To view...	Do this...
Product documentation	On the IBM Tealeaf portal, go to ? > Product Documentation .
IBM Tealeaf Knowledge Center	On the IBM Tealeaf portal, go to ? > Product Documentation and select <i>IBM Tealeaf Customer Experience in the ExperienceOne Knowledge Center</i> .
Help for a page on the IBM Tealeaf Portal	On the IBM Tealeaf portal, go to ? > Help for This Page .
Help for IBM Tealeaf CX PCA	On the IBM Tealeaf CX PCA web interface, select Guide to access the <i>IBM Tealeaf CX PCA Manual</i> .

Available documents for IBM Tealeaf products

The following table is a list of available documents for all IBM Tealeaf products:

Table 6. Available documentation for IBM Tealeaf products.

IBM Tealeaf products	Available documents
IBM Tealeaf CX	• <i>IBM Tealeaf Customer Experience Overview Guide</i> • <i>IBM Tealeaf CX Client Framework Data Integration Guide</i> • <i>IBM Tealeaf CX Configuration Manual</i> • <i>IBM Tealeaf CX Cookie Injector Manual</i> • <i>IBM Tealeaf CX Databases Guide</i> • <i>IBM Tealeaf CX Event Manager Manual</i> • <i>IBM Tealeaf CX Glossary</i> • <i>IBM Tealeaf CX Installation Manual</i> • <i>IBM Tealeaf CX PCA Manual</i> • <i>IBM Tealeaf CX PCA Release Notes</i>

Table 6. Available documentation for IBM Tealeaf products (continued).

IBM Tealeaf products	Available documents
IBM Tealeaf CX	• <i>IBM Tealeaf CX RealTime Viewer Client Side Capture Manual</i> • <i>IBM Tealeaf CX RealTime Viewer User Manual</i> • <i>IBM Tealeaf CX Release Notes</i> • <i>IBM Tealeaf CX Release Upgrade Manual</i> • <i>IBM Tealeaf CX Support Troubleshooting FAQ</i> • <i>IBM Tealeaf CX Troubleshooting Guide</i> • <i>IBM Tealeaf CX UI Capture j2 Guide</i> • <i>IBM Tealeaf CX UI Capture j2 Release Notes</i>
IBM Tealeaf cxImpact	• <i>IBM Tealeaf cxImpact Administration Manual</i> • <i>IBM Tealeaf cxImpact User Manual</i> • <i>IBM Tealeaf cxImpact Reporting Guide</i>
IBM Tealeaf cxConnect	• <i>IBM Tealeaf cxConnect for Data Analysis Administration Manual</i> • <i>IBM Tealeaf cxConnect for Voice of Customer Administration Manual</i> • <i>IBM Tealeaf cxConnect for Web Analytics Administration Manual</i>
IBM Tealeaf cxOverstat	<i>IBM Tealeaf cxOverstat User Manual</i>
IBM Tealeaf cxReveal	• <i>IBM Tealeaf cxReveal Administration Manual</i> • <i>IBM Tealeaf cxReveal API Guide</i> • <i>IBM Tealeaf cxReveal User Manual</i>
IBM Tealeaf cxVerify	• <i>IBM Tealeaf cxVerify Installation Guide</i> • <i>IBM Tealeaf cxVerify User's Guide</i>
IBM Tealeaf cxView	<i>IBM Tealeaf cxView User's Guide</i>
IBM Tealeaf CX Mobile	• <i>IBM Tealeaf CX Mobile Android Logging Framework Guide</i> • <i>IBM Tealeaf Android Logging Framework Release Notes</i> • <i>IBM Tealeaf CX Mobile Administration Manual</i> • <i>IBM Tealeaf CX Mobile User Manual</i> • <i>IBM Tealeaf CX Mobile iOS Logging Framework Guide</i> • <i>IBM Tealeaf iOS Logging Framework Release Notes</i>

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing IBM Corporation North Castle Drive Armonk, NY
10504-1785 U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing Legal and Intellectual Property Law IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Bay Area Lab 1001 E Hillsdale Boulevard Foster City, California 94404 U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Trademarks

IBM, the IBM logo, and ibm.com[®] are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Privacy Policy Considerations

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. A cookie is a piece of data that a web site can send to your browser, which may then be stored on your computer as a tag that identifies your computer. In many cases, no personal information is collected by these cookies. If a Software Offering you are using enables you to collect personal information through cookies and similar technologies, we inform you about the specifics below.

Depending upon the configurations deployed, this Software Offering may use session and persistent cookies that collect each user's user name, and other personal information for purposes of session management, enhanced user usability, or other usage tracking or functional purposes. These cookies can be disabled, but disabling them will also eliminate the functionality they enable.

Various jurisdictions regulate the collection of personal information through cookies and similar technologies. If the configurations deployed for this Software Offering provide you as customer the ability to collect personal information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for providing notice and consent where appropriate.

IBM requires that Clients (1) provide a clear and conspicuous link to Customer's website terms of use (e.g. privacy policy) which includes a link to IBM's and Client's data collection and use practices, (2) notify that cookies and clear gifs/web beacons are being placed on the visitor's computer by IBM on the Client's behalf along with an explanation of the purpose of such technology, and (3) to the extent required by law, obtain consent from website visitors prior to the placement of cookies and clear gifs/web beacons placed by Client or IBM on Client's behalf on website visitor's devices

For more information about the use of various technologies, including cookies, for these purposes, See IBM's Online Privacy Statement at: <http://www.ibm.com/privacy/details/us/en> section entitled "Cookies, Web Beacons and Other Technologies."



Printed in USA